Back în plan

/\* sa se simuleze parcurgerea unei table de sah de catre un cal

care pleaca dintr-o pozitie (x,y) a.i. sa parcurga fiecare camp

o singura data

\*/

#include <fstream>

#include<stdlib.h>

usingnamespacestd;

ifstream f("cal.in");

ofstream g("cal.out");

intdx[]={0, -2,-1,1,2,2,1,-1,-2};// 0 pentru a pleca de pe pozitia 1

intdy[]={0,1,2,2,1,-1,-2,-2,-1 };

//dx, dymodificarile coordonatelor de deplasare cf.regulilor de sah, fata de punctul curent

intnd=8;// nd= numarul de directii posibile

int a[11][11],x,y,ns=0,n; // ns numarsolutii

// x,y coordonatele de plecare

// tabla de sah se va considera in matricea din liniile/coloanele 2:8

// linia/coloana 0,1 vor avea valoarea -1 =ocupat

voidBordareMatrice()

{int i;

for(i=0;i<=n+3;i++)

{a[0][i]=-1;a[1][i]=-1;

a[i][0]=-1;a[i][1]=-1;

a[n+2][i]=-1;a[i][n+2]=-1;

a[n+3][i]=-1;a[i][n+3]=-1;

}

}

voidafis()

{ int i,j;

ns++;

g<<"solutianumarul "<<ns<<endl;

for(i=2;i<=n+1;i++)

{ for(j=2;j<=n+1;j++)

g<<a[i][j]<<" ";

g<<endl;

}

exit(EXIT\_SUCCESS);

// daca se doreste o singura solutie

// dupaafisareexit(EXIT\_SUCCESS)

}

void cal(int x, int y, int k)

{ // nr de ordine parcurgere tabla

int i;

a[x][y]=k;// marcare trecere

if(k==n\*n) afis();

else

{ // deplasare in directiile posibile

for(i=1;i<=nd;i++)

if(a[x+dx[i]][y+dy[i]]==0)// nu am parcurs acest camp

{

cal(x+dx[i],y+dy[i],k+1 );

}

a[x][y]=0;// cobor in stiva si anulez campul marcat

}

}

intmain()

{ int k;

f>>n;//n maxim 8 dimensiunea tablei de sah

// pentru n=5 si x=2, y=2 genereazasolutie

// am tinut cont de bordarea matricei (1,1) devine (2,2)

f>>x>>y;

BordareMatrice();

k=1;

cal(x,y,k);

if(ns==0) g<<"problema nu admite solutie pentru datele de intrare";

f.close();

g.close();

return 0;

}

Ex. pentru n=5 si x=2, y=2 va genera

Solutia numarul  1  
1    20    17    12    3      
16    11    2    7    18      
21    24    19    4    13      
10    15    6    23    8      
25    22    9    14    5